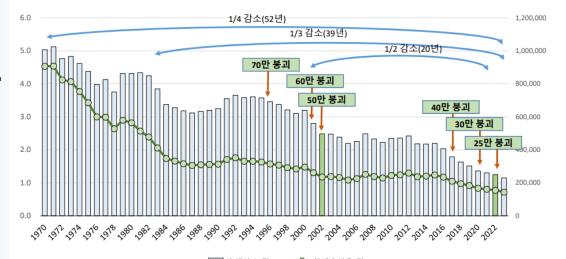
AI 회귀 분석을 이용한 대한민국 <mark>한</mark>구 예측

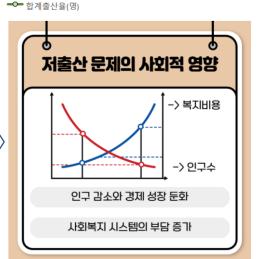
4조 (송봉환, 김윤준, 채수병)

- 목 차
- 1 기획 의도
 - 2 개발 목표
 - 3 주요 기능 설명
 - 4 업무 분장
 - 5 화면 설계서
 - 6 빅데이터 분석서
 - 7 소프트웨어 아키텍쳐
 - 8 향후 개발 계획
 - 9 프로젝트 수행 소감

기획 의도



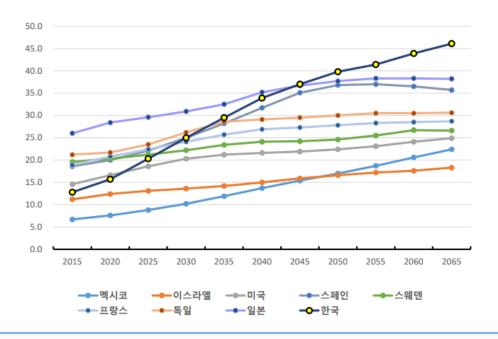




저출산 문제 인식 각인!

- 1. 인구 감소와 경제 성장 둔화
- 2. 사회복지 시스템의 부담 증가
- 3. 사회적 구조 변화와 세대 갈등
- 4. 교육 및 주택 문제

기획 의도



- 그러나 30년 안에 세계에서 가장 높은 고령화 수준에 이를 것
- 우리나라의 고령화는 수준이 아닌 속도의 문제
- 가까운 미래에 빠른 고령화는 사회 전반에 걸쳐 경험하지 않은 결과들을 만들어 낼 것

개발 목표 ①계획 수립

- 프로젝트 기간 : 2024년 12월 30일~ 2025년 1월 24일
- 예측 목표 : 2025년부터 2075년까지를 목표로 인구 예측을 시행합니다.
- 출생인구, 사망인구, 이주인구 데이터 등 여러 데이터를 활용.
- 데이터는 연도별로 제공되어야 하며, 과거 10년 이상의 데이터를 기본으로 합니다.
- 과거 데이터 분석을 통한 미래 인구를 예측합니다.

②사망인구 및 출생인구

사망인구 및 출생인구를 예측하는 인공지능 모델의 구현



미래 인구를 예측하는 인공지능 모델의 생성으로 보다 쉽게 미래 인구 체계를 볼 수 있다

개발 목표

③인구 피라미드

데이터를 이용하여 출산 인구 및 사망 인구 예측 10년 단위의 연령대별 인구량 시각화



예측한 데이터를 활용하여 피라미드 모양으로 구현 연도별 변화되는 저출산 및 고령화 문제 인식!

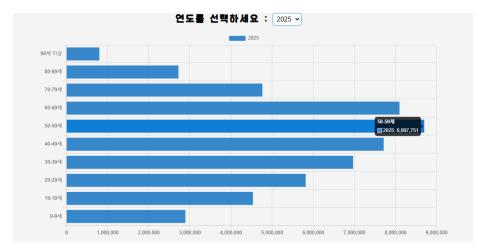
④지역 소멸 예측

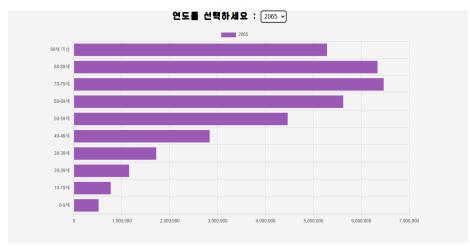
예측했던 출산 인구 및 사망 인구 데이터와 23년도 인구를 기준으로 지역 소멸 위험도 수치화



5년마다 지역 소멸 위험도 예측하여 게이지로 시각화 표현 저출산 및 고령화로 인한 지역 소멸 위험 인식!

주요 기능 설명

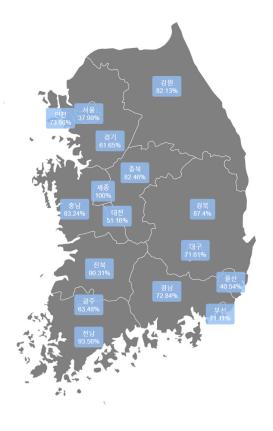




2025년부터 2075년까지 10년 단위 인구 생태계 시각화

- 시대의 변화에 따른 인구 생태계의 변화 확인
- 커서로 확인 가능한 연령대별 인구 수치 시각화

주요 기능 설명





2025년부터 2075년까지 5년 단위 도시 위치에 소멸도 수치화

- 도시 클릭시 도시별 정보 제공
- 지역 소멸 위험도 수치에 따른 게이지 색상 변화
- 년도 및 도시에 따른 인구 예측 서비스 제공



업무 분장

	작업 내용	1주차			2주차					3주차				4주차						
송 봉 환	기획 구상 및 스케줄 작성																			
	요구사항 정의서 작성																			
	HTML																			
	CSS																			
	데이터 수집 및 전처리																			
	python을 이용한 코드 작성																			
	기능 구현 (미래 출생인구 및 사망인구 예측 → 지역별 인구 예측 지역 소멸 위험도 예측)																			
	기능 html에 삽입 및 최종 점검																			
	프로젝트 보고서 및 개인 발표 자료 작성																			

1) 메인 화면

(1) ® 그 해 한국은





About Us



고망화한 65세 이상의 고양 인구 비용이 크게하게 증가하는 사회적 현상을 의미합니다 2013년 기준으로 일본 독일 이탈리아가 노인인구 20% 이상인 최고립사회에 이미 진입했다. 이러한 추세는 다른 국가에도 이어져 203이년에는 전 세계적으로 초고입사회에 불인한 국가가 물러 34개국에 당할 것으로 전환하였다. 이처럼 많은 국가들이 고검화문제에 작면해 있지만, 그 중 고검화가 가장 심각한 곳은 우리나라이다 2제개발합력기구(OECD)의 보고서 역시 '한국은 그 동안 가장 젊은 나라인지만. 항호 SOM 이내 가장 높은 나라로 변화한 것이다 '라고 전략하였다 UN 보고서 역시 우리나라가 2026년에는 초고령사회로 진입할 것으로 전망하고 있다

우리나라 함께 출산물(15~49세의 출산 가능한 여성이 정생 남는 자녀의 수)은 1960년 6,0에 목박하였다 그러나 인구증가를 경제성장의 약제 요인으로 인식한 박장의 중권에 의해 국가적 차원에서 가족계획사업이 적극적으로 실시되고, 1960~1970년대의 산업화 및 이성의 경제활동 참여 증가 등의 요인이 맞물리면서 출산물이 급격히 감소하기 시작하였다. 1970년 4.5명으로 감소한 우리나라 합계층산을은 1984년 2.06명으로 현상 유지가 가능한 인구대체 수준인 2.1명 하래로 떨어졌다 가족계획사업이 종급된 호안 199이네 이오에도 축산물은 지속적으로 각소하여 2000년 1,47명, 2010년 1,23명으로 지극히 낮은 수준을 유지하여 현재 우리나라는 '초 저출산시대를 맞이하고 있다. 2014년 현재 우리나라 함께 총산물은 1,21에 불과하여, 미국 1,86, 일본 1,42보다 낮다.

이는 2대한테게받기구(OECD) 34개 회원국 중에서 가장 낮은 소중이다.



2) 화면 설명

로고

- 클릭 시 메인 화면으로 이동 기능
 - 사이트 내 모든 페이지에 표시
 - 한국을 뜻하는 K K 상단에 아이를 들고 있는 부모가 있음

네비게이션 바

- 클릭 시 각 기능 페이지로 이동
- 사이트 내 모든 페이지에 표시
- 해당 사이트는 검은색으로 표시
 - 비해당 사이트는 회색으로 표시, 커서 올라가면 검은색으로 변화

사이트 설명

- 고령화와 저출산에 대한 문제 설명
- 고령화에 대한 설명
 - 저출산에 대한 설명

퓨터

- 상단 4채널 클릭 시 메인 화면으로 이동 기능
- 좌측 그림 클릭 시 인구 피라미드 이동 기능
 - 우측 그림 클릭 시 지역 소멸 위험도 이동 기능
 - 하단 클릭시 학원 홈페이지 이동 기능



빅데이터 분석서



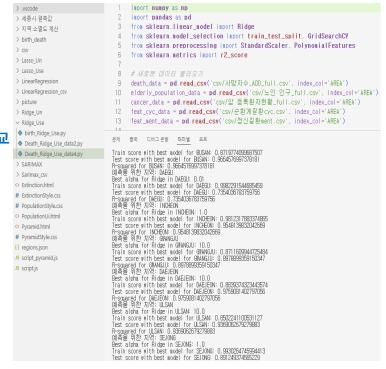
GridSearchCV 및 PolynomialFeatures 사용 또는 미사용하여
Ridge와 Lasso를 사용하여 트레이닝 값과 테스트 값 비교하였더니
Ridge에서 GridSearchCV 및 PolynomialFeatures을 사용했을 때의 값이 가장 이상적인 값이 확인됨

빅데이터 분석서

```
1 import numpy as np
> .vscode
                                    2 import pandas as pd
> 세종시 결측값
                                      from sklearn.linear model import Ridge
> 지역 소멸도 계산
                                    4 from sklearn.model_selection import train_test_split, GridSearchCV
> birth_death
                                    5 from sklearn.preprocessing inport StandardScaler, PolynomialFeatures
> csv
                                       from sklearn.metrics import r2 score
> Lasso Un
                                   8 # 새로운 데이터 불러오기
 LinearRegression
                                   9 death_data = pd.read_csv('csv/사망자수_ADD_full.csv', index_col='AREA')
> LinearRegression_csv
                                   10 elderly_population_data = pd.read_csv('csv/노인 인구_full.csv', index_col='AREA')
                                   11 cancer data = pd.read csv('csv/암 등록환자현황 full.csv', index col='AREA')
> picture
> Ridge_Un
                                  문제 출력 디버그콘솔 터미널 포트

∨ Ridge_Use

birth Ridge Use.py
                                  예측을 위한 지역: SFOIL
                                                                                                                                 비교
                                  Best alpha for Ridge in SEOUL: 10.0
Death_Ridge_Use_data2.pg
                                   Frain score with best model for SEOUL: 0.8146099635945379
                                  Test score with best model for SEOUL: 0.6872558618034179
Death Ridge Use data4.pv
                                  R-squared for SEOUL: 0.6872558618D34179
예측을 위한 지역: BUSAN
> SARIMAX
                                  Best alpha for Ridge in BUSAN: 1.0
> Sarimax csv
                                  Train score with best model for BUSAN: 0.9370471405747071
                                  Test score with best model for BUSAN: 0.8864771983805922
                                  R-squared for BUSAN: 0.8864771983805922
# ExtinctionStyle.css
                                  예측을 위한 지역: DAEGU
# PopulationStyle.css
                                  Best alpha for Ridge in DAEGU: 1.0
                                  Train score with best model for DAEGU: 0.9317574247876198
O PopulationUi.html
                                  Test score with best model for DAEGU: 0.8134156345311729
O Pyramid.html
                                  R-squared for DAEGU: 0.8134156345311729
                                  예측을 위한 지역: INCHEON
# PyramidStyle.css
                                  Best alpha for Ridge in INCHEON: 1.0
                                  Train score with best model for INCHEON: 0.97609456211446
{} regions.ison
                                  Test score with best model for INCHEON: 0.9377123480711996
JS script_pyramid.js
                                  R-squared for INCHEON: 0.9377123480711996
                                  예측을 위한 지역: GWANGJU
                                  Best alpha for Ridge in GWANGJU: 1.0
                                  Train score with best model for GMANGJU: 0.9302125390896611
                                  Test score with best model for GWANGJU: 0.8887476337911743
                                  R-squared for GWANGJU: 0.8887476337911743
                                  예측을 위한 지역: DAEJEON
                                  Best alpha for Ridge in DAEJEON: 10.0
                                  Train score with best model for DAEJEON: 0.8709190401330991
                                  Test score with best model for DAEJEON: 0.7605129109093722
                                  R-squared for DAEJEON: 0.7605129109093722
                                  예측을 위한 지역: ULSAN
                                 Best alpha for Ridge in ULSAN: 10.0
                                  Train score with best model for ULSAN: 0.8501648031285355
                                  Test score with best model for ULSAN: 0.7184253896928672
```



위에서 설명한 것과 같이 Ridge로 사망자 인구를 머신러닝으로 돌렸으나 원하는 테스트 점수가 나오지 않음. 데이터 수의 부족이라 생각해서 데이터 2개를 더 수집하여 재 테스트 시도 완료.

데이터 2개를 넣은 값과 데이터 4개를 넣은 값의 차이가 확인됨.

스프트웨어 아키텍쳐





- 머신러닝 모델 생성
- 데이터 불러오기 / 저장, 전처리



- 홈페이지 구현
- 각 버튼 기능 구현
- 데이터 시각화



- 인구 소멸 게이지 제작
- Chart.js를 이용한 데이터 차트 시각화

향후 개발 계획

1. 로그인 기능 추가

홈페이지인 만큼 로그인 기능을 추가하여 새로운 기능도 경험해보고 사용자에게 다양한 기능을 선사함

2. 보다 정확한 머신 러닝

자료를 좀 더 찾아서 대입하고, 개선하여 보다 발전된 머신 러닝 개발

3. 홈페이지 기능 개선

현재는 서비스 웹인 느낌 보다 홈페이지 느낌이 강함, 기능 발전 및 개선을 통한 서비스 제공 예정

4. 기능 추가 개발

현재 개발 된 기능보다 다른 기능에 대한 아이디어 제시 및 실현 예정

프로젝트 수행 소감

나오지 않았으며, 중간에 데이터 부족으로 인한 문제 해결을 위해 회의를 통하여 데이터 재수집을 하였습니다. 목표한 기능을 전부 수행하였지만, 데이터 몇 개를 넣어서 미래 인구수를 예측하는 기능에 있어서 완벽하다고 할 순 없다고 생각이 듭니다. 하지만 그 데이터를 가지고 보다 완벽하기 위해 노력을 했고, 기능을 만들어 시각화에 성공했다는 경험이 값지다고 생각합니다. 계획도 세밀하게 짜고 못 할 것 같지 않았는데 막상 프로젝트를 진행하다 보니 딜레이 되는 부분도 있고, 생각대로, 원하는 방향대로 흘러가지 않아 많은 어려움이 있었습니다.

더 좋은 기능 구현을 위해 앞으로 노력을 많이 해야겠다는 생각이 자주 들었습니다.

코드 작업 부분에서 에러가 자주 발생하여 문제 해결을 하였고, 트레이닝 값과 테스트 값이 원하는 수치가

조장으로써 팀원들 간의 화합을 가장 중요하다고 생각하는데 팀원 중에 이런 비슷한 강의를 6개월 거쳐 수료를 했는데도 불구하고 프로젝트를 실패한 인원이 있었습니다. 자존감도, 자신감도 바닥이어서 프로젝트 시작하는 날부터 중간중간까지도 못하겠다, 능력이 안된다 등의 부정적인 말을 하곤 했는데 그떄마다 할 수 있다고, 계획대로 하면 충분히 구현 가능하다고, 자신감을 불어넣어 주고 처음부터 끝까지 할 일을 주며 맡은 바 최선을 다할 수 있게 이끌었습니다. 중간에 버리고 싶다고도 생각이 들었지만, 이번마저도 실패를 겪는다면 좌절을 하지 않을까 라는 생각에 놓질 못했습니다 프로젝트가 끝나고 그 친구한테 "봐봐 하면 할 수 있잖아"라고 자신감을 불어 넣어준 게 아직도 생각이 듭니다 이 팀으로 결과물을 만들어서 다행이고, 뿌듯합니다

감사합니다